

图文 032、案例实战：百万级用户的在线教育平台，如何基于G1垃圾回收器优化性能（上）？

1878 人次阅读 2019-08-01 07:00:00

[详情](#) [评论](#)

案例实战：百万级用户的在线教育平台

如何基于G1垃圾回收器优化性能（上）？

给大家推荐一套质量极高的Java面试训练营课程：



作者是中华石杉，石杉老哥是我之前所在团队的 Leader，骨灰级的技术神牛！

大家可以点击下方链接，了解更多详情，并进行试听：

[21天互联网Java进阶面试训练营（分布式篇）](#)

重要说明：

最近不少同学留言反馈，说希望建立一个微信群，供大家进行JVM专栏的学习交流。

这个提议非常好，不过管理微信群是一件挺费时的事儿，我平时工作较忙，实在抽不出时间来进行群管理。

正好石杉老哥的面试训练营建了微信交流群，并且还请了不少一线大厂的助教。

因此跟石杉老哥商量了一下，决定厚着脸皮“鸠占鹊巢”。购买了我JVM专栏的小伙伴，可以加入石杉老哥的微信群，在群里讨论交流技术。

目录：

- 1、案例背景引入
- 2、系统核心业务流程分析
- 3、系统的运行压力
- 4、本文总结与下文预告

1、案例背景引入

咱们接下来用一个真实的案例来给大家聊聊G1垃圾回收器的一些性能优化的手段和背后的原理

其实核心思想跟之前我们讲过的“ParNew+CMS”的垃圾回收器组合的优化思想是类似的，但是因为G1的运行原理有一些不一样的地方，所以说在优化上会略有不同。

首先我们来说说案例的背景，这是一个**百万级注册用户的在线教育平台**，主要目标用户群体是几岁到十几岁的孩子，注册用户大概是几百万的规模，日活用户规模大概在几十万。

系统的业务流程其实也不复杂，而且我们可以排除掉一些选课、排课、浏览课程详情以及付费购买之类的低频的行为。

为什么这么说呢？大家可以想一下，一个在线教育平台，本来用户也不是特别多，他不是一个电商平台，不会说每天每个人都会进来浏览幼儿课程详情。

所以一般业务流程就是，有人进来浏览一下，考虑一段时间，然后下定决心了给自己孩子报名一个在线的英语课程，或者数学课程之类的。

所以其实其他的普通用户浏览课程详情、下单付费、选课排课，这些都是绝对的低频行为，我们几乎不用考虑到系统的运行中去，可以暂时忽略掉。

那么对于这样的一个系统，他最关键的高频行为是什么呢？

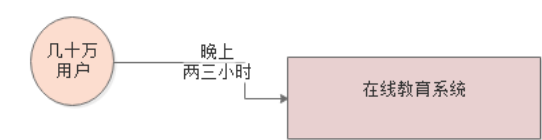
上课！

也就是说，大家从这个系统的本质来思考一下，这样的一个平台，他的使用人群是幼儿园的孩子到中小学的孩子

他们平时白天都要上学，一般也就是晚上放学之后到八九点钟的样子，是最活跃使用这个平台的时候，还有就是周末也是最活跃使用这个平台的时候。

这里尤为关键的需要注意的，就是每天晚上那两三小时的高峰时期，几乎你可以认为每天几十万日活用户（那些小孩儿）都会集中在这个时间段来平台上在线课程，比如青少年英语课，或者数学课之类的。

所以这个晚上两三小时的时间段里，将会是平台每天绝对的高峰期，而且白天几乎没什么流量，可能99%的流量都集中在晚上，如下图所示。



2、系统核心业务流程分析

接着我们来明确一下，这样的一个系统，孩子们在上课的时候主要高频使用的是哪些功能呢？

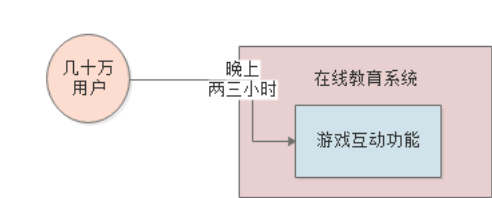
其实非常的简单，现在如果大家家里有孩子，平时对一些在线教育APP有一定了解的话，应该知道现在在线教育APP都会尤为主打的是互动环节。

给大家举个例子，比如说给五六岁的孩子上的幼儿英语课，大家觉得，还会跟20年前一样，给他们放非常枯燥的“李雷和韩梅梅”的情节么？然后机械式的去跟读？

那肯定不是了！现在尤为强调的是在欢快的娱乐游戏中进行教学，让孩子们快乐的学习英语、数学之类学科的知识。

所以说，在那几十万用户晚上高峰时间段使用系统上课的时候，尤为核心的业务流程，就是大量的游戏互动环节

通过游戏互动让孩子们感兴趣，愿意学，而且通过游戏强互动让他们保持注意力，促使他们对学习到的东西进行输出，提升学习的效果，大家看下图。



也就是说，这个游戏互动功能，一定会承载用户高频率、大量的互动点击

比如在完成什么任务的时候必须要点击很多的按钮，频繁的进行互动，然后系统后台需要接收大量的互动请求，并且记录下来用户的互动过程和互动结果。

系统得记录下来用户完成了多少个任务，做对了几个，做错了几个，诸如此类的。

3、系统的运行压力

现在我们开始来分析一下这个系统运行时候对内存使用产生的一个压力

其实核心点就是搞明白在晚上两三小时高峰期内，每秒钟会有多少请求，每个请求会连带产生多少对象，占用多少内存，每个请求要处理多长时间。

首先我们来分析一下晚上高峰期内几十万用户同时在线使用平台，每秒钟会产生多少请求？

我们可以大致来估算一下，比如说晚上3小时高峰期内有总共60万活跃用户，平均每个用户大概会使用1小时左右来上课，那么每小时大概会有20万活跃用户同时在线学习。

这20万活跃用户因为需要进行大量的互动操作，所以大致可以认为是每分钟进行1次互动操作，一小时内会进行60次互动操作

那么20万用户在1小时内会进行1200万次互动操作，平均到每秒钟大概是3000次左右的互动操作，这是一个很合理的数字。

那么每秒钟要承载3000并发请求，根据经验来看，一般系统的核心服务需要部署5台4核8G的机器来抗住是差不多的，每台机器每秒钟抗个600请求，这个压力可以接受，一般不会导致宕机的问题。

那么每个请求会产生多少个对象呢？

一次互动请求不会有太复杂的对象，他主要是记录一些用户互动过程的，可能会跟一些积分之类的东西有关联

大家如果玩儿过在线教育APP都知道，每次你完成一个活动，一般会给你累加一些对应的“XX币”，“XX宝石”之类的东西。

所有大致估算一下，一次互动请求大致会连带创建几个对象，占据几KB的内存，比如我们就认为是5KB吧那么一秒600请求会占用3MB左右的内存。

4、本文总结与下文预告

基于百万用户在线教育平台的G1垃圾回收优化案例，分为上下两文

本文先按照惯例，把**整个系统的背景、核心业务流程、高峰运行压力、机器部署、每秒请求压力、每秒内存使用压力**，都给分析了出来。

接着下文就会基于这个每秒内存使用压力，结合G1垃圾回收器的运行原理，来给大家分析在这个压力之下，G1垃圾回收机制会如何来运行，这个过程中可能会产生哪些问题，如何对G1的一些参数进行基本的优化来调整垃圾回收的性能。

但是G1本身运行机制极为复杂，所以真的要搞懂他很多优化的细节，必须结合后面**几十个不同的案例**，不同的场景来说明，一个案例也仅仅能把部分优化方式给说清楚而已。

End

专栏版权归公众号**狸猫技术窝**所有

未经许可不得传播，如有侵权将追究法律责任

如何加群？

1、添加微信号：Giotto1245（微信名：Jarvis）